IN THE CLAIMS

Please cancel claims 1, 5 and 9.

Please amend the claims as follows.

1. (Cancelled)

2. (Currently Amended) ~~The apparatus of claim 1 further comprising~~ An apparatus comprising:

at least one processor;

a memory coupled to the at least one processor;

a database table residing in the memory;

a cardinality estimator residing in the memory and executed by the at least one processor, the cardinality estimator estimating cardinality of an intermediate dataset that satisfies a query to the database table in a manner that accounts for data skew in the database table;

a query optimizer residing in the memory that uses the estimated cardinality from the cardinality estimator to optimize the query; and

a frequent values list residing in the memory that contains a list of values in the database table, each value having a corresponding frequency, wherein the cardinality estimator estimates the cardinality of the intermediate dataset by determining whether a frequency corresponding to a value exceeds a predetermined threshold, and if the frequency exceeds the predetermined threshold, accounting for the corresponding value, and if the frequency does not exceed the predetermined threshold, using a formula to estimate the cardinality of the intermediate dataset, the formula accounting for data skew in the database table by subtracting the frequency of all values above the predetermined threshold in the frequent values list that satisfy the query from the total number of rows in the ~~database table~~ intermediate dataset.

3. (Original) The apparatus of claim 2 wherein the cardinality estimator estimates the cardinality Ca' of the intermediate dataset using the formula:

$$Ca' = P + M(1-(1-\frac{1}{M})^{Y})$$

where

M = Ca - (P+Q)

P = number of distinct values in the frequent values list above the predetermined threshold that satisfy the query;

Q = number of distinct values in the frequent values list above the predetermined threshold that do not satisfy the query;

Ca = cardinality of the database table;

Y = X - Fi;

X = number of rows in the intermediate dataset; and

Fi = sum of frequencies of values in the frequent values list above the predetermined threshold that satisfy the query.

4. (Previously Presented) An apparatus comprising:

at least one processor;

a memory coupled to the at least one processor;

a database table residing in the memory;

a frequent values list residing in the memory that contains a list of values in the database table, each value having a corresponding frequency; and

a cardinality estimator residing in the memory and executed by the at least one processor, the cardinality estimator estimating cardinality of an intermediate dataset for a query to the database table using the following formula:

$$Ca' = P + M(1 - (1 - \frac{1}{M})^{Y})$$

where

M = Ca - (P+Q)

P = number of distinct values in the frequent values list above a predetermined threshold that satisfy the query;

Q = number of distinct values in the frequent values list above the predetermined threshold that do not satisfy the query;

Ca = cardinality of the database table;

Y = X - Fi;

X = number of rows in the intermediate dataset; and

Fi = sum of frequencies of values in the frequent values list above the predetermined threshold that satisfy the query; and

a query optimizer residing in the memory that uses the estimated cardinality of the intermediate dataset to optimize the query.

5. (Cancelled)

6. (Currently Amended) ~~The method of claim 5 wherein step (B) includes the steps of~~:

A method for optimizing a query to a database table by estimating cardinality of an intermediate dataset that results from processing the query, the method comprising the steps of:

(A) evaluating the query;

(B) estimating cardinality of the intermediate dataset using a formula that accounts for data skew in the database table by performing the steps of:

selecting a value in a frequent values list that contains a list of values in the database table, each value having a corresponding frequency;

if the selected value has a corresponding frequency that exceeds a predetermined threshold, incrementing the cardinality estimate by one; and

if the frequency does not exceed the predetermined threshold, using a formula to estimate the cardinality of the intermediate dataset, the formula accounting for data skew in the database table by subtracting the frequency of all values above the predetermined threshold in the frequent value table that satisfy the query from the total number of rows in the ~~database table~~ intermediate dataset; and

(C) using the cardinality estimate in step (B) to optimize the query.

7. (Previously Presented) The method of claim 6 wherein the cardinality estimator estimates the step of estimating the cardinality Ca' of the intermediate dataset in step (B) uses the formula:

$$Ca' = P + M(1-(1-\frac{1}{M})^{Y})$$

where

M = Ca - (P+Q)

P = number of distinct values in the frequent values list above the predetermined threshold that satisfy the query;

Q = number of distinct values in the frequent values list above the predetermined threshold that do not satisfy the query;

Ca = cardinality of the database table;

Y = X - Fi;

X = number of rows in the intermediate dataset; and

Fi = sum of frequencies of values in the frequent values list above the predetermined threshold that satisfy the query.

8. (Previously Presented) A method for optimizing a query to a database table by estimating cardinality of an intermediate dataset that results from processing the query on a database table, the method comprising the steps of:

(A) evaluating the query;

(B) estimating the cardinality Ca' of the intermediate dataset using the formula:

$$Ca' = P + M(1 - (1 - \frac{1}{M})^{Y})$$

where

M = Ca - (P+Q)

P = number of distinct values in the frequent values list above a predetermined threshold that satisfy the query;

Q = number of distinct values in the frequent values list above the predetermined threshold that do not satisfy the query;

Ca = cardinality of the database table;

Y = X - Fi;

X = number of rows in the intermediate dataset; and

Fi = sum of frequencies of values in the frequent values list above the predetermined threshold that satisfy the query; and

(C) using the cardinality estimate in step (B) to optimize the query.

9-11 (Cancelled)

12. (Currently Amended) ~~The program product of claim 9~~ A computer-readable program product comprising:

(A) cardinality estimator estimating cardinality of an intermediate dataset that satisfies a query to a database table in a manner that accounts for data skew in the database table, wherein the cardinality estimator evaluates a frequent values list that contains a list of values in the database table, each value having a corresponding frequency, wherein the cardinality estimator estimates the cardinality of the intermediate dataset by determining whether a frequency corresponding to a value exceeds a predetermined threshold, and if the frequency exceeds the predetermined threshold, accounting for the corresponding value, and if the frequency does not exceed the predetermined threshold, using a formula to estimate the cardinality of the intermediate dataset, the formula accounting for data skew in the database table by subtracting the frequency of all values above the predetermined threshold in the frequent values list that satisfy the query from the total number of rows in the ~~database table~~ intermediate dataset;

(B) a query optimizer that uses the estimated cardinality from the cardinality estimator to optimize the query; and

(C) recordable media bearing the cardinality estimator and the query optimizer.

13. (Previously Presented) The program product of claim 12 wherein the cardinality estimator estimates the cardinality Ca' of the intermediate dataset using the formula:

$$Ca' = P + M(1 - (1 - \frac{1}{M})^{Y})$$

where

M = Ca - (P+Q)

P = number of distinct values in the frequent values list above the predetermined threshold that satisfy the query;

Q = number of distinct values in the frequent values list above the predetermined threshold that do not satisfy the query;

Ca = cardinality of the database table;

Y = X - Fi;

X = number of rows in the intermediate dataset; and

Fi = sum of frequencies of values in the frequent values list above the predetermined threshold that satisfy the query.

14. (Currently Amended) A computer-readable program product comprising:

(A) a cardinality estimator that estimates cardinality of an intermediate dataset for a query to a database table using the following formula:

$$Ca' = P + M(1 - (1 - \frac{1}{M})^{Y})$$

where

M = Ca - (P+Q)

P = number of distinct values in a frequent values list above a predetermined threshold that satisfy the query;

Q = number of distinct values in the frequent values list above the predetermined threshold that do not satisfy the query;

Ca = cardinality of the database table;

Y = X - Fi;

X = number of rows in the intermediate dataset; and

Fi = sum of frequencies of values in the frequent values list above the predetermined threshold that satisfy the query;

(B) a query optimizer that uses the estimated cardinality from the cardinality estimator to optimize the query; and

(C) recordable media bearing the cardinality estimator and the query optimizer.

15. (Cancelled)

16. (Cancelled)

STATUS OF THE CLAIMS

Claims 1-16 were originally filed in this patent application. In response to a first office action dated 05/19/2006, an amendment was filed on 08/18/2006 that cancelled claims 10-11 and 15-16, and amended claims 1-2, 4-9 and 12-14. In the pending final office action, claims 2, 3, 6, 7, 12 and 13 were rejected under 35 U.S.C. §112, second paragraph. Claims 9 and 12-14 were rejected under 35 U.S.C. §101. Claims 1, 5 and 9 were rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent Application Publication 2004/0059743 to Burger in view of Applicant's Admitted Prior Art (AAPA). Claims 4 and 8 were allowed. In this amendment, claims 1, 5 and 9 have been cancelled, and claims 2, 6, 12 and 14 have been amended. Claims 2-4, 6-8 and 12-14 are currently pending.